IDOR: PERM_EDIT_USERS allows modifying any user's notification subscriptions (incomplete fix of CVE-2025-48472)

Moderate **freescout-help-desk** published **GHSA-f489-qxv6-gvgg** 2 weeks ago

Package

freescout-help-desk/freescout

Affected versions	Patched versions
<1.8.217	1.8.217

Description

Summary

A user holding the `PERM_EDIT_USERS` permission (intended for general user-profile editing) can read and modify the notification subscriptions of **any other user**, including admins, by sending a single POST request. This is a sibling of CVE-2025-48472's notification authorization bypass — the prior fix did not cover this code path.

A non-admin attacker can silently disable an admin's email/browser/mobile notifications, suppressing security alerts and conversation-assignment notices.

Affected versions

`<= 1.8.216` (tested on commit `e6fe63e71f37b5b9683454edcf5a40e4288c0e36`).

Root cause

`app/Http/Controllers/UsersController.php`, `notificationsSave()` lines 397-406:

```
public function notificationsSave($id, Request $request)
{
    $user = User::findOrFail($id);
    $this->authorize('update', $user);                          // shared 'update' policy
    Subscription::saveFromArray($request->subscriptions, $user->id);
    \Session::flash('flash_success_floating', __('Notifications saved successfully'));
    return redirect()->route('users.notifications', ['id' => $id]);
}
```

`app/Policies/UserPolicy.php`, `update()` lines 53-63:

```
public function update(User $user, User $modelUser)
{
    return $user->isAdmin() || $user->hasPermission(User::PERM_EDIT_USERS);
}
```

The `update` policy is intended for profile editing — it gates fields like `first_name`, `phone`, `timezone`. The `notificationsSave` endpoint reuses this same policy, so any `PERM_EDIT_USERS` holder can write to any user's `subscriptions` rows. There is no per-target ownership check.

Reproduction (live RBAC 3-scenario test)

Test environment: 3 users — admin (id=1, role=ROLE_ADMIN), editor (id=3, role=ROLE_USER, `permissions={"10":1}` = PERM_EDIT_USERS), agent (id=2, role=ROLE_USER, no permissions). Admin starts with 8 notification subscriptions.

#	Setup	Status	Result
A	Editor `GET /users/notifications/1` (admin's settings)	**200**	Admin's notification page renders — non-admin can read it
B	Editor `POST /users/notifications/1` with empty `subscriptions[]`	**302**	Admin's 8 subscriptions cleared to **0**
C	Plain agent (no `PERM_EDIT_USERS`) `GET /users/notifications/1`	**403**	Correctly denied — confirms permission check exists, just over-broad

```
JAR=/tmp/editor.cookies
# Login editor first

CSRF=$(curl -sS -b "$JAR" "http://target/users/notifications/1" \
  | sed -n 's/.*name="csrf-token" content="\([^"]*\)".*/\1/p' | head -1)

curl -sS -b "$JAR" -X POST "http://target/users/notifications/1" \
  --data-urlencode "_token=$CSRF"
```

```
# -> 302 Location: /users/notifications/1
# Admin's subscription rows: 8 -> 0
```

DB observation:

```
mysql> SELECT COUNT(*) FROM subscriptions WHERE user_id=1;
+----------+
| COUNT(*) |
+----------+
|        0 |       -- was 8 before editor's POST
+----------+
```

Impact

- **Disable admin security alerts** (login alerts, conversation assignments, mailbox events)
- **Operational disruption**: silently change notification routing
- **Defense-in-depth degradation**: admin loses awareness of suspicious actions
- Variant of CVE-2025-48472 — same root *pattern* (over-broad `update` policy applied to non-update operations) at a different endpoint.

Limited blast radius: only notification preferences (not password/email/role). The original CVE-2025-48472 fix should be extended to cover this and any other `notificationsSave`-shaped endpoints.

Suggested fix

Enforce self-only access unless caller is admin:

```
--- a/app/Http/Controllers/UsersController.php
+++ b/app/Http/Controllers/UsersController.php
@@ -397,7 +397,10 @@
 public function notificationsSave($id, Request $request)
 {
     $user = User::findOrFail($id);
-    $this->authorize('update', $user);
+    if (!auth()->user()->isAdmin() && (int) auth()->id() !== (int) $id) {
+        abort(403);
+    }
     Subscription::saveFromArray($request->subscriptions, $user->id);
```

Long-term: split `UserPolicy@update` into `updateProfile`, `updateNotifications`, `updatePassword` (already separate via the hard self-only check at `passwordSave():597`), and audit every `authorize('update', $user)` call site for over-broad delegation.

Distinction from CVE-2025-48472

CVE-2025-48472 fixed the original notification authz bypass at a different endpoint. This advisory covers the `notificationsSave` endpoint that was missed by the original patch. Same root pattern, sibling code path.

Disclosure

Discovered 2026-04-17 / 2026-04-18 by automated security analysis (web-vuln-agent). 90-day disclosure window: public disclosure planned 2026-07-17.

Severity

Moderate 5.4 / 10

CVSS v3 base metrics

Attack vector	Network
Attack complexity	Low
Privileges required	Low
User interaction	None
Scope	Unchanged
Confidentiality	None
Integrity	Low
Availability	Low

Learn more about base metrics

CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:U/C:N/I:L/A:L

CVE ID

CVE-2026-41903

Weaknesses

▶ CWE-863

Credits

 **whatisproblem** Reporter